

Please type a plus sign (+) inside this box → ☐

PTO/SB/05 (2/98)
Approved for use through 09/30/00. OMB 0651-0032
Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

UTILITY PATENT APPLICATION TRANSMITTAL

(Only for new nonprovisional applications under 37 CFR 1.53(b))

Attorney Docket No.	042390.P8995
First Inventor or Application Identifier	Narsing K. Vijayrao
Title	METHOD AND APPARATUS FOR IMPROVING THE PERFORMANCE OF A
Express Mail Label No.	EM560889914US

APPLICATION ELEMENTS
See MPEP chapter 600 concerning utility patent application contents

ADDRESS TO: Assistant Commissioner for Patents
Box Patent Application
Washington, DC 20231

- ☒ Fee Transmittal Form (e.g. PTO/SB/17)
(Submit an original, and a duplicate for fee processing)
- ☒ Specification *Total Pages* **14**
(preferred arrangement set forth below)
 - Descriptive title of the Invention
 - Cross References to Related Applications
 - Statement Regarding Fed sponsored R & D
 - Reference to Microfiche Appendix
 - Background of the Invention
 - Brief Summary of the Invention
 - Brief Description of the Drawings (if filed)
 - Detailed Description
 - Claim(s)
 - Abstract of the Disclosure
- ☒ Drawing(s) (35 U.S.C. 113) *Total Sheets* **4**
- ☐ Oath or Declaration *Total Pages* **3**
 - ☒ Newly executed (original copy)
 - ☐ Copy from a prior application (37 CFR 1.63(d))
(for continuation/divisional with Box 16 completed)
 - ☐ **DELETION OF INVENTOR(S)**
Signed statement attached deleting inventor(s) named in the prior application, see 37 CFR 1.63(d)(2) and 1.33(b).

- ☐ Microfiche Computer Program (Appendix)
- ☐ Nucleotide and/or Amino Acid Sequence Submission
(if applicable, all necessary)
 - ☐ Computer Readable Copy
 - ☐ Paper Copy (identical to computer copy)
 - ☐ Statement verifying identity of above copies

ACCOMPANYING APPLICATION PARTS

- ☒ Assignment Papers (cover sheet & document(s))
- ☐ 37 CFR 3.73(b) Statement ☐ Power of Attorney
(when there is an assignee)
- ☐ English Translation Document (if applicable)
- ☐ Information Disclosure Statement (IDS)/PTO - 1449 ☐ Copies of IDS Citations
- ☐ Preliminary Amendment
- ☒ Return Receipt Postcard (MPEP 503)
(Should be specifically itemized)
- ☐ *Small Entity Statement filed in prior application, Statement(s) ☐ Status still proper and desired
- ☐ Certified Copy of Priority Document(s)
(if foreign priority is claimed)
- ☐ Other:

*NOTE FOR ITEMS 1 & 13: IN ORDER TO BE ENTITLED TO PAY SMALL ENTITY FEES, A SMALL ENTITY STATEMENT IS REQUIRED (37 C.F.R. § 1.27), EXCEPT IF ONE FILED IN A PRIOR APPLICATION IS RELIED UPON (37 C.F.R. § 1.28).

16. If a CONTINUING APPLICATION, check appropriate box, and supply the requisite information below and in a preliminary amendment:

☐ Continuation ☐ Divisional ☐ Continuation-in-part (CIP) of prior application No: _____ / _____
Prior application Information: Examiner _____ Group/Art Unit: _____

For CONTINUATION or DIVISIONAL APPS only: The entire disclosure of the prior application, from which an oath or declaration is supplied under Box 4b, is considered a part of the disclosure of the accompanying continuation or divisional application and is hereby incorporated by reference. The incorporation can only be relied upon when a portion has been inadvertently omitted from the submitted application parts.

17. CORRESPONDENCE ADDRESS

☐ Customer Number of Bar Code Label ☐ (Insert Customer No. or Attach bare code label here) or ☒ Correspondence address below

Name	BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP				
Address	12400 Wilshire Boulevard, Seventh Floor				
City	Los Angeles	State	California	Zip Code	90025
Country	U.S.A.	Telephone	(310) 207-3800	Fax	(310) 820-5988

Name (Print/Type) Thomas M. Coester, Reg. No. 39,637

Signature

Thomas M. Coester

Date

06/27/00

Burden Hour Statement: This form is estimated to take 0.2 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Box Patent Application, Washington, DC 20231.

Docket No.: 042390.P8995
Express Mail No.: EM560889914US

UNITED STATES PATENT APPLICATION

FOR

**METHOD AND APPARATUS FOR IMPROVING THE PERFORMANCE OF
A FLOATING POINT MULTIPLIER ACCUMULATOR**

Inventors:
Narsing K. Vijayrao
Chi Keung Lee
Sudarshan Kumar

Prepared By:

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN
12400 Wilshire Blvd., 7th Floor
Los Angeles, California 90025-1026
(310) 207-3800

0042390.P8995

METHOD AND APPARATUS FOR IMPROVING THE PERFORMANCE OF A FLOATING POINT MULTIPLIER ACCUMULATOR

BACKGROUND

Field of the Invention

The present invention relates to the field of computer processors and floating point mathematical support in computer processors. More specifically, this invention relates to improving the performance of a floating point multiplier accumulator.

Background

Computers are ubiquitous in modern society. Computers are regularly used for complex mathematical renderings required by modern computer graphics demands as well as traditional accounting, architecture and other specialized mathematically intensive application programs. Mathematical computations which require vary large numbers, require high precision, and/or include complex mathematical equations are referred to as floating point calculations. When programming software, floating point numbers are used when performing floating point calculations. Floating point numbers are commonly defined as having three parts: a sign, a significand (also known as a mantissa), and an exponent. A well known standard that sets a framework for how floating point numbers and calculations should be implemented is I.E.E.E. standard 754 (1985, reaffirmed 1990), the Standard for Binary Floating point Arithmetic, available from the Institute of Electrical and Electronics Engineers, Inc., 445 Hoes Lane, Piscataway, New Jersey 08855-1331 (the I.E.E.E. Floating point Standard).

Floating point support has been implemented in a number of ways with processors. In earlier personal computers, a floating point co-processor was optionally available to be installed with and to assist a processor in handling floating point calculations (*e.g.*, Intel Corporation provided a Numeric Processor Extension chip named the 8087 to accompany the widely used 8086 processor). As personal computers have evolved, processors have

incorporated floating point capability within a processor by including one or more floating point units in a processor.

Traditionally, only specialized scientific and accounting application programs accessed a processor's floating point capabilities. However, today, colorful graphic and multimedia images are in widespread use in, for example, internet web pages, architectural software applications, computer games, and animation creation programs. Further, the use of Digital Video Disks (DVD) and the impending on-demand download of video presentations such as movies will cause increased usage of floating point capabilities of processors in computers and more specialized viewing devices. In all of these uses, images are stored in various compressed or encoded formats. The more detailed and higher resolution an image is, the more floating point calculations are needed to process (*i.e.*, decompress or decode) and render the image for display on a monitor or other image generating device. As the use of graphic images has become popular and continues to grow, the use of a processor's floating point mathematical capabilities has been increasing. In addition, many other computer and processor uses, including use for audio processing, are also contributing to an increased use of a processor's floating point mathematical capabilities. To accommodate these and other needs, and to meet the ever growing demand for increased floating point performance, the floating point capability of processors is continually evolving. Any incremental increase in floating point throughput will increase the throughput of processors, computers, viewing devices, and any other systems utilizing the floating point capabilities of a processor.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 depicts a prior art floating point multiplier accumulator.

Figure 2 depicts the flow of actions taken according to a prior art method of implementing a floating point multiplier accumulator.

Figure 3 depicts one embodiment of a floating point multiplier accumulator according to the present invention.

Figure 4 depicts the flow of actions taken according to the method of implementing a floating point multiplier accumulator according to the present invention.

DETAILED DESCRIPTION

Floating point numbers are implemented in a limited bit space, often in 32, 64, and 128 bit widths. Whenever computations are performed on floating point numbers, the bit width of the significand may be exceeded. For ease of reference, the bits beyond the designated significand bit width are described herein as "fallout" bits. Depending on the selected rounding mode and the fallout bits, the result of the computation may have to be modified by rounding. The discussion and invention herein involves the processing and computation of the significand portion of floating point numbers. For ease of reference, the term floating point number is used even though the significand is what is being acted on.

A floating point multiplier accumulator (FMAC) by definition receives as input three floating point numbers, A, B and C, and produces $(A \times B + C)$ as a result. In traditional, prior art systems, an FMAC processes $(A \times B + C)$ to create what is known as SUM and CARRY, adds SUM and CARRY, shifts or normalizes the result, and then performs rounding. According to the present invention, to increase performance of an FMAC, when the SUM and CARRY are added together, the resulting sum, the resulting sum plus one, and the resulting sum plus two are computed in parallel so that the value necessitated by any needed rounding is computed in advance of the traditional rounding step. According to the present invention, the appropriate result is then chosen according to the rounding mode and normalized. In one embodiment, this method reduces the number of clock cycles needed for an FMAC to complete its execution by one. That is, in one embodiment, one clock cycle is saved by the use of the apparatus and method of the present invention. Although one clock cycle alone is a small amount of time, with the ever increasing use of floating point calculations and concomitant reliance on the floating point capabilities of processors, a nontrivial increase

in overall processor performance results. In addition, only a minimal increase of on-chip hardware is required to accomplish the method and achieve the performance improvement.

Figure 1 depicts a prior art floating point multiplier accumulator.

Floating point multiplier accumulator (FMAC) 100 receives as input floating point numbers A, B, and C. The FMAC outputs the properly rounded result of “(A x B + C)” based on the rounding mode. Multiplier 102 receives A, B and C and outputs what is known as SUM and CARRY. Propagate, kill, generate (PKG) generator 104 is coupled to multiplier 102 and receives SUM and CARRY. Using SUM and CARRY, the PKG generator produces (1) P, the product of A and B, (2) G, the sum of A and B, and (3) K, the product of the one’s complement of A and the one’s complement of B. That is:

$$P = A + B$$

$$G = A \times B$$

$$K = \text{complement}(A) \times \text{complement}(B)$$

Adder 106 is coupled to the PKG generator and receives as input P, K and G and uses P, K and G to determine the sum of SUM and CARRY. Leading zero anticipator (LZA) 108 is also coupled to the PKG generator and receives P, K and G from the PKG generator. Normalization shifter 110 receives as input the result from adder 106 and the position of the decimal point from leading zero anticipator 108. Rounding unit 112 receives the normalized result from normalization shifter 110, and then increments, decrements or leaves unaffected the normalized result, depending on the rounding mode. The rounding mode is determined by, in one embodiment, reading the contents of a well-known register in the processor. The rounding mode is one of the rounding modes provided for in the I.E.E.E. Floating Point Standard. Pursuant to this standard, the rounding mode may be round toward positive infinity, round toward negative infinity, round toward zero, and round toward nearest. In one prior art embodiment, rounding in the form of incrementing or decrementing requires at least one clock cycle. In this prior art implementation, the total time for the FMAC processing

includes the time needed to sequentially perform SUM plus CARRY addition and in adder 106 then perform rounding in rounding unit 112, if needed.

Figure 2 depicts the flow of actions taken according to a prior art method of implementing a floating point multiplier accumulator. The prior art FMAC receives three floating point numbers designated as A, B and C as input, as shown in block 200. $A \times B + C$ is then computed in SUM and CARRY form, as shown in block 204. P, K and G are then generated using SUM and CARRY, as shown in block 208. SUM and CARRY are then added using P, K, and G, as shown in block 210. The position of where the decimal point is located is then determined by the leading zero anticipator, as shown in block 212. The result, $A \times B + C$, is then shifted according the result of the leading zero determination, as shown in block 220. The shifted result is then incremented or decremented, if needed, according to the rounding mode, as shown in block 224. In this prior art method, the adding SUM and CARRY and the determination of the leading zero may be performed in parallel. As mentioned above, in this prior art method, the total time for FMAC processing includes the time required to sequentially perform multiplication, generate P, K and G, perform addition, normalize and to perform rounding, if needed.

Figure 3 depicts one embodiment of a floating point multiplier accumulator according to the present invention. Floating point multiplier accumulator (FMAC) 300 receives as input floating point numbers A, B, and C. Multiplier 304 receives A, B and C, produces $A \times B + C$ and outputs the result in SUM and CARRY form. Propagate, kill, generate (PKG) generator 308 is coupled to multiplier 304 and receives the SUM and CARRY as input. The PKG generator produces P, K, and G using SUM and CARRY, and may be the same PKG generator as in the prior art. Adder 310, plus-one 312, plus-two-er 314, and leading zero anticipator 316 are coupled to PKG generator 308 and all receive P, K, and G as input. Adder 310 uses P, K and G to add SUM and CARRY. Plus-one 312 uses P, K and G to add SUM and CARRY and increment the resulting sum by one. Plus-two-er 314 uses P, K and G to add SUM and CARRY and increment the resulting sum by two. Leading zero

anticipator (LZA) 316 determines the location of the decimal point of the result by computing a leading zero position. Multiplexor 320 receives the result of each of adder 310, plus-one-r 312, and plus-two-r 314 and selects which of the results is appropriate responsive to a control signal received from rounding control 322. Rounding control 322 issues an appropriate signal responsive to the rounding mode and the output of leading zero anticipator 316. The rounding mode may be obtained from, in one embodiment, a register in the processor. Normalization shifter 326 receives as input the appropriate sum selected by multiplexor 320. According to this method, the prior art steps of adding followed by rounding are effectively achieved in parallel by the computations accomplished by adder 310, plus-one-r 312, plus-two-r 314, multiplexor 320, and rounding control 322. That is, the prior art rounding hardware is replaced with multiplexor 320, and rounding control 322 which complete their execution in less time than traditional rounding, and, thus take the resulting FMAC requires less time to complete its computation. This results in a relatively small increase in hardware required on the processor while increasing floating point computation throughput by, in one embodiment, one clock cycle.

Figure 4 depicts the flow of actions taken according to the method of implementing a floating point multiplier accumulator according to the present invention. The FMAC receives three floating point numbers designated as A, B and C as input, as shown in block 400. $A \times B + C$ is then computed in SUM and CARRY form, as shown in block 404. P, K and G are then generated, as shown in block 408. PKG generator 408 may be the same as the PKG generator used in prior art systems. Four operations then occur in parallel: (1) SUM and CARRY are added together using P, K and G, as shown in block 410; (2) SUM and CARRY are added together using P, K and G and the resulting sum is then incremented by one, as shown in block 412; (3) SUM and CARRY are added together using P, K and G and the resulting sum is then incremented by two, as shown in block 414; and (4) the location of the decimal point is made by determination of the position of leading zeros, as shown in block 416. The appropriate result of blocks 410, 412, and 414 is then

selected according to the rounding mode and the position of the leading zero, as shown in block 420. In one embodiment, the rounding mode may be determined by examination of a register in the processor in which the FMAC resides. The result is then normalized according the outcome of the leading zero determination, as shown in block 430. By determining the result of the addition, the addition with incrementing by one, and the addition with incrementing by two, all of the possible results responsive to the rounding mode are predetermined. To appropriately round the result, the appropriate value produced by blocks 410, 412 and 414 in parallel is selected, eliminating the time consuming two step adding and rounding sequence taught in the prior art. As rounding is often necessary, the selection of the result responsive to the rounding mode saves time and increases throughput, particularly when performing numerous floating point computations.

Although the method and apparatus described above are resident in a processor, the method may also be implemented in software and microcode in those processors that allow and provide for such an implementation. Such software may reside within a processor, in cache memory, in random access memory, etc. In addition, such software may be read by the processor during boot up as part of a basic input output system (BIOS) or similar startup sequence and may be read from a hard disk, floppy disk, stick memory device, programmable read only memory (PROM), flash memory, or any other kind of machine readable medium.

In the foregoing specification, the invention has been described with reference to specific embodiments thereof. It will, however, be evident that various modifications and changes can be made thereto without departing from the broader spirit and scope of the invention as set forth in the appended claims. The specification and drawings are, accordingly, to be regarded in an illustrative rather than a restrictive sense. Therefore, the scope of the invention should be limited only by the appended claims.

CLAIMS

What is claimed is:

- 1 1. A significand portion of a floating point multiply accumulator
2 (FMAC) comprising:
3 a multiplier receiving a first input significand, a second input significand,
4 and a third input significand;
5 a propagate, kill, generate generator (PKG generator) coupled to the
6 multiplier;
7 an adder, a plus-one-r, a plus-two-er and a leading zero anticipator (LZA)
8 each coupled to the PKG generator;
9 a rounding control unit coupled to the LZA;
10 a multiplexor coupled to each of the adder, the plus-one-r, the plus-two-er
11 and the rounding control unit; and
12 and a normalization shifter coupled to the multiplexor and the LZA.
- 1 2. The significand portion of Claim 1 wherein the multiplier outputs a
2 sum value and a carry value.
- 1 3. The significand portion of Claim 2 wherein the PKG generator
2 computes a propagate value (P), a kill value (K) and a generate value (G) based
3 on the sum value and the carry value.
- 1 4. The significand portion of Claim 3 wherein in parallel the adder
2 adds the sum value and the carry value using P, K, and G; the plus-one-r adds the
3 sum value and the carry value using P, K, and G and increments by one; and the
4 plus-two-er adds the sum value and the carry value using P, K, and G and
5 increments by two.
- 1 5. The significand portion of Claim 4 wherein the LZA computes in
2 parallel with the adder, the plus-one-r, and the plus-two-er.

1 6. The significand portion of Claim 1 wherein the rounding control
2 unit reads a rounding mode from a register in a processor in which the FMAC
3 resides.

1 7. The significand portion of Claim 1 wherein the normalization
2 shifter and the rounding control unit each receive a leading zero position
3 indication from the LZA.

1 8. The significand portion of Claim 1 wherein the multiplexor
2 produces an output result responsive to the rounding control unit.

1 9. A floating point multiply accumulator (FMAC) comprising:
2 a multiplier;
3 a propagate, kill, generate generator (PKG generator) to produce a
4 propagate value (P), a kill value (K) and a generate value (G) coupled to the
5 multiplier;
6 an adder, a plus-one, a plus-two-er and a leading zero anticipator (LZA)
7 each coupled to the PKG generator in parallel;
8 a rounding control unit coupled to the LZA and coupled to a multiplexor,
9 the multiplexor outputting a result from one of the adder, the plus-one, and the
10 plus-two-er responsive to the rounding control unit; and
11 and a normalization shifter coupled to the multiplexor and the LZA.

1 10. The FMAC of Claim 9 wherein the multiplier produces a product of
2 a first floating point number and a second floating point number added to a third
3 floating point number as a sum value and a carry value.

1 11. The FMAC of Claim 10 wherein in parallel the adder adds the sum
2 value and the carry value using P, K, and G; the plus-one adds the sum value
3 and the carry value using P, K, and G and increments by one; and the plus-two-er
4 adds the sum value and the carry value using P, K, and G and increments by two.

1 12. The FMAC of Claim 9 wherein the rounding control unit outputs a
2 select signal to the multiplexor based on a rounding mode and the decimal point
3 position.

1 13. The FMAC of Claim 9 wherein the normalization shifter
2 normalizes based on the decimal point position.

1 14. A floating point multiply accumulator (FMAC) comprising:
2 a means for multiplying a first significand and a second significand and
3 adding a third significand to produce a sum value and a carry value;
4 a means for computing a propagate value, a kill value, and a generate
5 value coupled to the means for multiplying;
6 a first means for adding the sum value to the carry value;
7 a second means for adding the sum value to the carry value and
8 incrementing by one;
9 a third means for adding the sum value to the carry value and
10 incrementing by two;
11 a means for determining a leading zero position, such that the first means
12 for adding, the second means for adding, the third means for adding, and the
13 means for determining are coupled in parallel to the means for computing;
14 a means for controlling responsive to the means for determining and a
15 rounding mode, the means for controlling further coupled to a means for
16 selecting, the means for selecting outputting a result from one of the first means
17 for adding, the second means for adding, and the third means for adding
18 responsive to the means for controlling; and
19 and a means for normalizing coupled to the means for selecting and the
20 means for determining.

1 15. The FMAC of Claim 14 wherein the means for controlling reads the
2 rounding mode from a register in a processor in which the FMAC resides.

1 16. The FMAC of Claim 14 wherein the means for normalizing is
2 responsive to the means for determining.

1 17. A method in a floating point multiply accumulator (FMAC)
2 comprising:
3 receiving a first floating point number, a second floating point number
4 and a third floating point number;
5 computing a product of the first floating point number and the second
6 floating point number and adding a third floating point number to produce a
7 sum value and a carry value;
8 computing a propagate value, a kill value and a generate value based on
9 the sum value and the carry value;
10 simultaneously adding the sum value to the carry value to create a first
11 result, adding the sum value to the carry value and incrementing by one to
12 create a second result, adding the sum value to the carry value and incrementing
13 by two to create a third result, and determining a decimal point position;
14 selecting one of the first result, the second result and the third result
15 responsive to a rounding mode and the decimal point position as a selected
16 result; and
17 normalizing the selected result based on the decimal point position.

1 18. The method of Claim 17 further comprising;
2 reading the rounding mode from a register in a processor in which the
3 FMAC resides.

1 19. The method of Claim 17 wherein normalizing comprises:
2 shifting the bits in the selected result.

1 20. The method of Claim 17 wherein the propagate value, the kill
2 value and the generate value are used by the adder, the plus-one and the plus-
3 two-er to compute the first result, the second result and the third result.

1 21. A machine readable medium containing instructions which, when
2 executed by a processor, cause a machine to perform operations comprising:
3 receiving a first floating point number, a second floating point number
4 and a third floating point number;
5 computing a product of the first floating point number and the second
6 floating point number and adding a third floating point number to produce a
7 sum value and a carry value;
8 computing a propagate value, a kill value and a generate value based on
9 the sum value and the carry value;
10 simultaneously adding the sum value to the carry value to create a first
11 result, adding the sum value to the carry value and incrementing by one to
12 create a second result, adding the sum value to the carry value and incrementing
13 by two to create a third result, and determining a decimal point position;
14 selecting one of the first result, the second result and the third result
15 responsive to a rounding mode and the decimal point position as a selected
16 result; and
17 normalizing the selected result based on the decimal point position.

1 22. The machine readable medium of Claim 21 containing instructions
2 which, when executed by a processor, cause the machine to perform further
3 operations comprising:
4 reading the rounding mode from a register in a processor in which the
5 FMAC resides.

1 23. The machine readable medium of Claim 21 wherein normalizing
2 comprises:
3 shifting the bits in the selected result.

ABSTRACT

A method and apparatus to increase the performance of a floating point multiplier accumulator (FMAC). The method comprises receiving three floating point numbers and computing a product of the first floating point number and the second floating point number and adding a third floating point number to produce a sum value and a carry value. A propagate value, a kill value and a generate value are then computed based on the sum value and the carry value. Simultaneously the sum value is added to the carry value to create a first result, the sum value is added to the carry value and incremented by one to create a second result, the sum value is added to the carry value and incremented by two to create a third result, and a decimal point position is determined. One of the first result, the second result and the third result is then selected responsive to a rounding mode and the decimal point position. The selected result is normalized based on the decimal point position. The apparatus comprises a multiplier with a propagate, kill, generate generator (PKG generator) coupled to it. An adder, a plus-one, a plus-two-er and a leading zero anticipator (LZA) are each coupled to the PKG generator in parallel. A rounding control unit is coupled to the LZA and coupled to a multiplexor that outputs a result from one of the adder, the plus-one, and the plus-two-er responsive to the rounding control unit. A normalization shifter is coupled to the multiplexor and the LZA.

FIG. 1
(PRIOR ART)

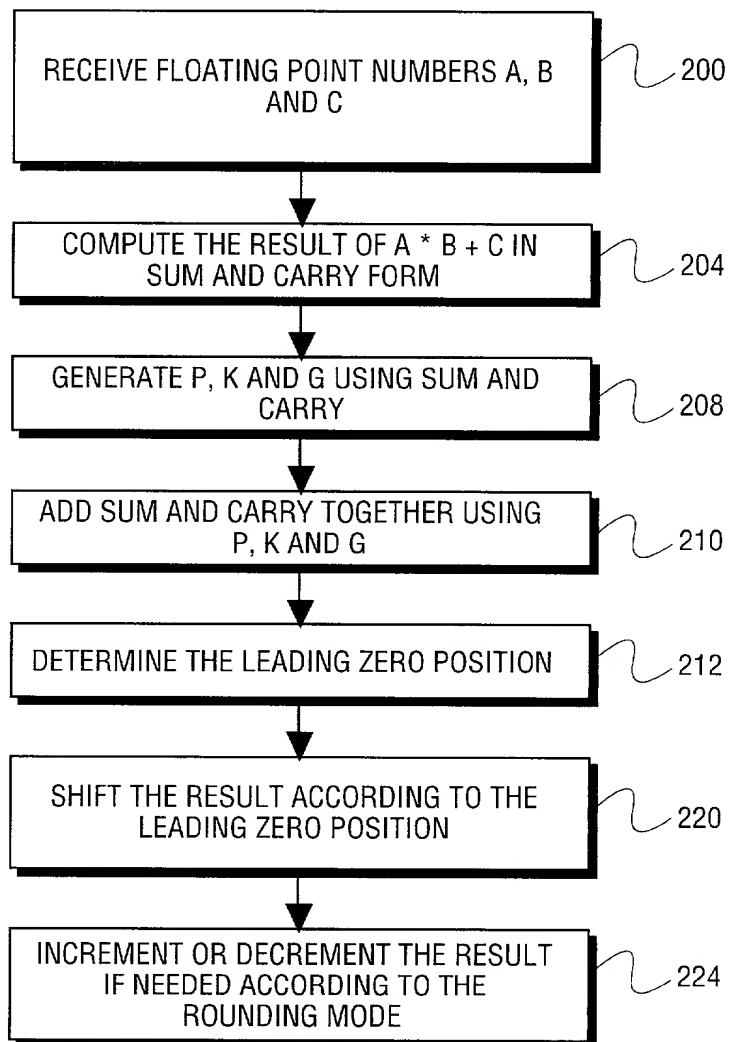


FIG. 2
(PRIOR ART)

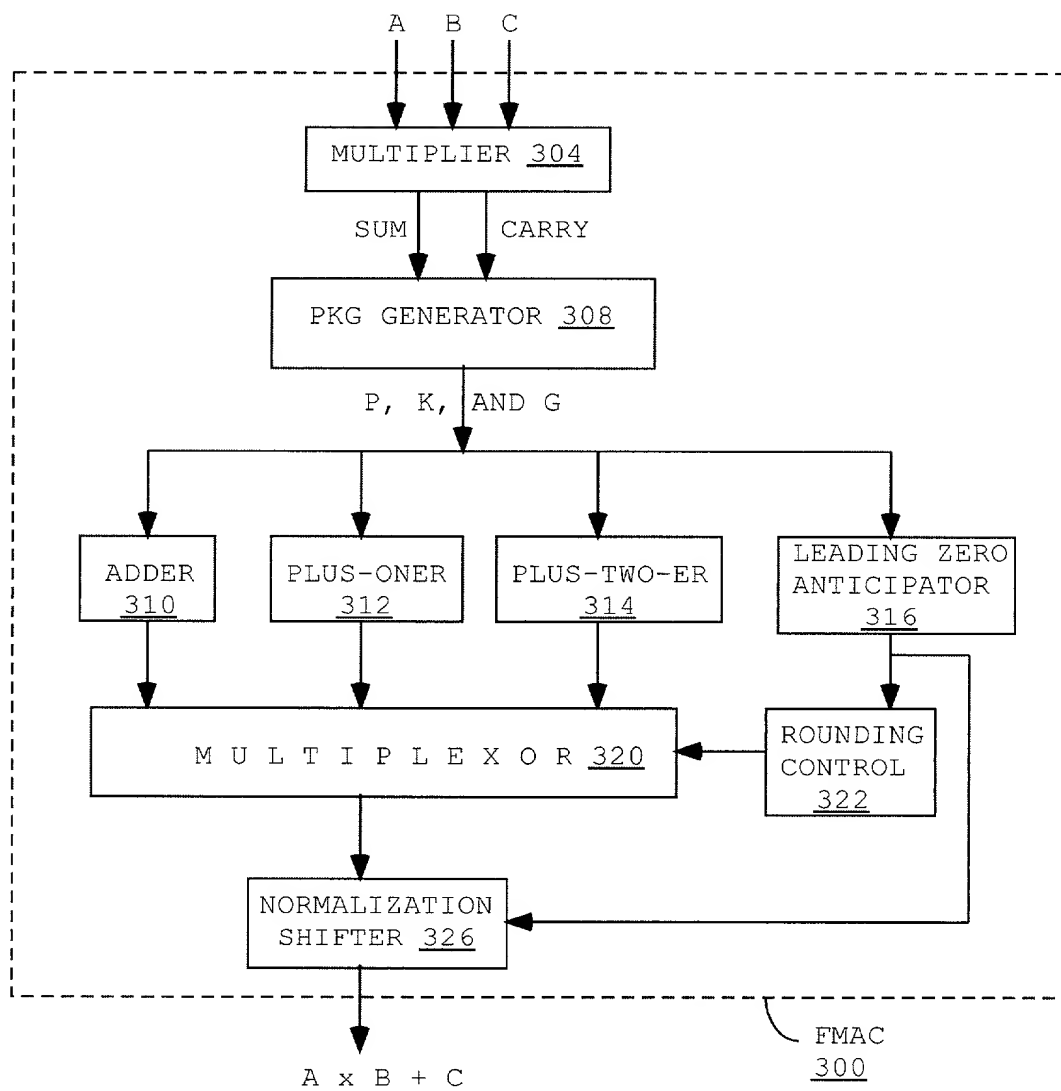


FIG. 3

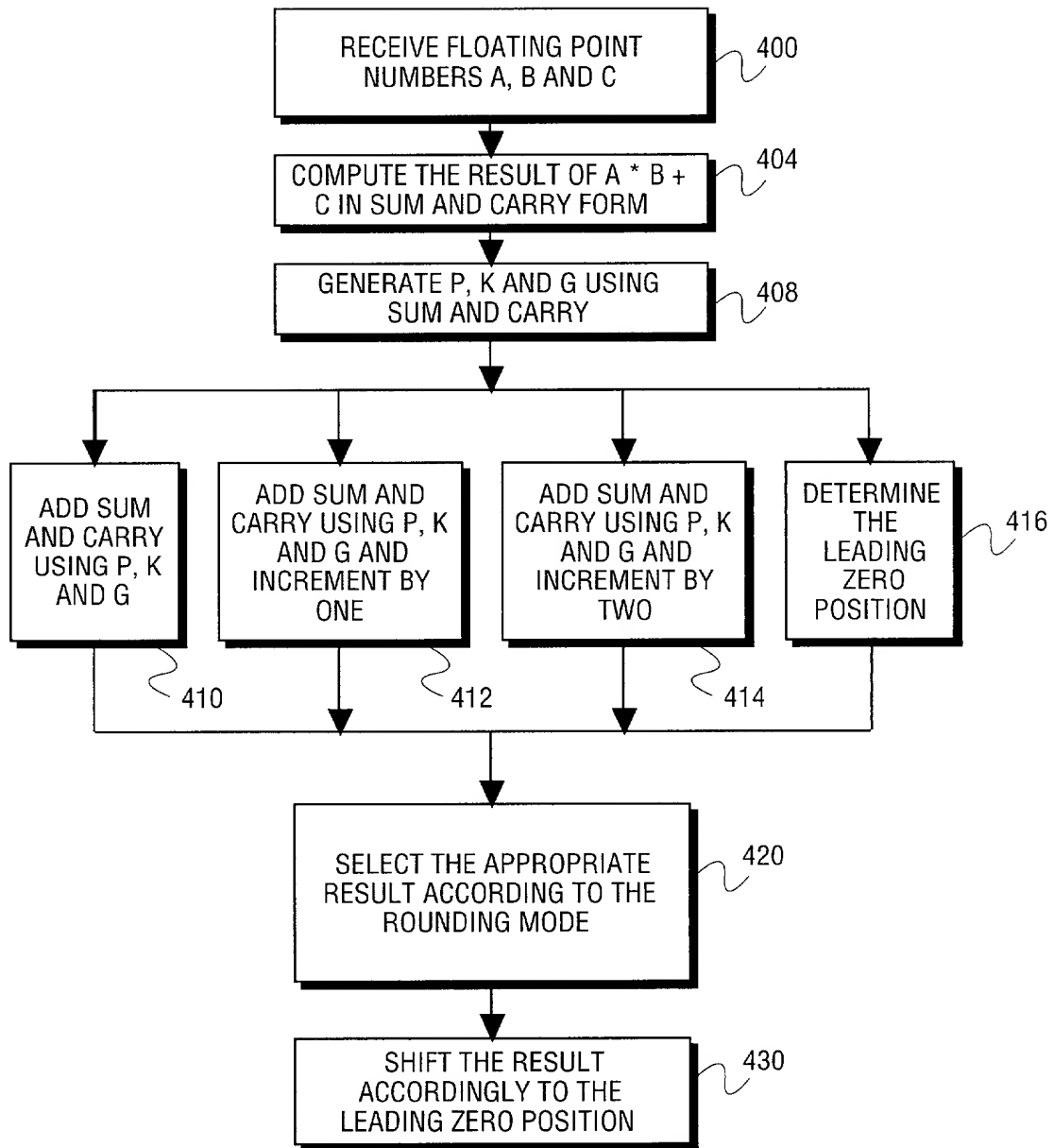


FIG. 4

DECLARATION AND POWER OF ATTORNEY FOR PATENT APPLICATION (FOR INTEL CORPORATION PATENT APPLICATIONS)

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below, next to my name.

I believe I am the original, first, and sole inventor (if only one name is listed below) or an original, first, and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled

METHOD AND APPARATUS FOR IMPROVING THE PERFORMANCE OF A FLOATING POINT MULTIPLIER ACCUMULATOR

the specification of which

☒ is attached hereto.
☐ was filed on _____ as
 United States Application Number _____
 or PCT International Application Number _____
 and was amended on _____
 (if applicable)

I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claim(s), as amended by any amendment referred to above. I do not know and do not believe that the claimed invention was ever known or used in the United States of America before my invention thereof, or patented or described in any printed publication in any country before my invention thereof or more than one year prior to this application, that the same was not in public use or on sale in the United States of America more than one year prior to this application, and that the invention has not been patented or made the subject of an inventor's certificate issued before the date of this application in any country foreign to the United States of America on an application filed by me or my legal representatives or assigns more than twelve months (for a utility patent application) or six months (for a design patent application) prior to this application.

I acknowledge the duty to disclose all information known to me to be material to patentability as defined in Title 37, Code of Federal Regulations, Section 1.56.

I hereby claim foreign priority benefits under Title 35, United States Code, Section 119(a)-(d), of any foreign application(s) for patent or inventor's certificate listed below and have also identified below any foreign application for patent or inventor's certificate having a filing date before that of the application on which priority is claimed:

Prior Foreign Application(s):

APPLICATION NUMBER	COUNTRY (OR INDICATE IF PCT)	DATE OF FILING (day, month, year)	PRIORITY CLAIMED
			<input type="checkbox"/> No <input type="checkbox"/> Yes
			<input type="checkbox"/> No <input type="checkbox"/> Yes
			<input type="checkbox"/> No <input type="checkbox"/> Yes

I hereby claim the benefit under Title 35, United States Code, Section 119(e) of any United States provisional application(s) listed below:

APPLICATION NUMBER	FILING DATE

I hereby claim the benefit under Title 35, United States Code, Section 120 of any United States application(s) listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code, Section 112, I acknowledge the duty to disclose all information known to me to be material to patentability as defined in Title 37, Code of Federal Regulations, Section 1.56 which became available between the filing date of the prior application and the national or PCT international filing date of this application:

APPLICATION NUMBER	FILING DATE	STATUS (ISSUED, PENDING, ABANDONED)

I hereby appoint BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN, a firm including: William E. Alford, Reg. No. 37,764; Farzad E. Amini, Reg. No. 42,261; Amy M. Armstrong, Reg. No. 42,265; Aloysius T. C. AuYeung, Reg. No. 35,432; William Thomas Babbitt, Reg. No. 39,591; Carol F. Barry, Reg. No. 41,600; Jordan Michael Becker, Reg. No. 39,602; Bradley J. Berezna, Reg. No. 33,474; Michael A. Bernadieu, Reg. No. 35,934; Roger W. Blakely, Jr., Reg. No. 25,831; Gregory D. Caldwell, Reg. No. 39,926; Ronald C. Card, Reg. No. 44,587; Thomas M. Coester, Reg. No. 39,637; Donna Jo Coningsby, Reg. No. 41,684; Michael Anthony DeSanctis, Reg. No. 39,957; Daniel M. De Vos, Reg. No. 37,813; Robert Andrew Diehl, Reg. No. 40,992; Matthew C. Fagan, Reg. No. 37,542; Tarek N. Fahmi, Reg. No. 41,402; Paramita Ghosh, Reg. No. 42,806; James Y. Go, Reg. No. 40,621; James A. Henry, Reg. No. 41,064; Willmore F. Holbrow III, Reg. No. 41,845; Sheryl Sue Holloway, Reg. No. 37,850; George W. Hoover II, Reg. No. 32,992; Eric S. Hyman, Reg. No. 30,139; William W. Kidd, Reg. No. 31,772; Sang Hui Kim, Reg. No. 40,450; Eric T. King, Reg. No. 44,188; Erica W. Kuo, Reg. No. 42,775; Michael J. Mallie, Reg. No. 36,591; Paul A. Mendonsa, Reg. No. 42,879; Darren J. Milliken, Reg. No. 42,004; Chun M. Ng, Reg. No. 36,878; Thien T. Nguyen, Reg. No. 43,835; Thinh V. Nguyen, Reg. No. 42,034; Dennis A. Nicholls, Reg. No. 42,036; Lisa A. Norris, Reg. No. 44,976; Daniel E. Ovanezian, Reg. No. 41,236; William F. Ryann, Reg. No. 44,313; James H. Salter, Reg. No. 35,668; William W. Schaal, Reg. No. 39,018; James C. Scheller, Reg. No. 31,195; Jeffrey S. Smith, Reg. No. 39,377; Maria McCormack Sobrino, Reg. No. 31,639; Stanley W. Sokoloff, Reg. No. 25,128; Judith A. Szepesi, Reg. No. 39,393; Vincent P. Tassinari, Reg. No. 42,179; Edwin H. Taylor, Reg. No. 25,129; Joseph A. Twarowski, Reg. No. 42,191; Lester J. Vincent, Reg. No. 31,460; Glenn E. Von Tersch, Reg. No. 41,364; John Patrick Ward, Reg. No. 40,216; Charles T. J. Weigell, Reg. No. 43,398; James M. Wu, Reg. No. 45,241; Steven D. Yates, Reg. No. 42,242; and Norman Zafman, Reg. No. 26,250; my attorneys; and Andrew C. Chen, Reg. No. 43,544; Justin M. Dillon, Reg. No. 42,486; and John F. Travis, Reg. No. 43,203; my patent agents, with offices located at 12400 Wilshire Boulevard, 7th Floor, Los Angeles, California 90025, telephone (310) 207-3800, and Alan K. Aldous, Reg. No. 31,905; Robert D. Anderson, Reg. No. 33,826; Joseph R. Bond, Reg. No. 36,458; Richard C. Calderwood, Reg. No. 35,468; Jeffrey S. Draeger, Reg. No. 41,000; Cynthia Thomas Faatz, Reg. No. 39,973; Sean Fitzgerald, Reg. No. 32,027; Seth Z. Kalson, Reg. No. 40,670; David J. Kaplan, Reg. No. 41,105; Charles A. Mirho, Reg. No. 41,199; Leo V. Novakoski, Reg. No. 37,198; Naomi Obinata, Reg. No. 39,320; Thomas C. Reynolds, Reg. No. 32,488; Kenneth M. Seddon, Reg. No. 43,105; Mark Seeley, Reg. No. 32,299; Steven P. Skabrat, Reg. No. 36,279; Howard A. Skaist, Reg. No. 36,008; Steven C. Stewart, Reg. No. 33,555; Raymond J. Werner, Reg. No. 34,752; Robert G. Winkle, Reg. No. 37,474; and Charles K. Young, Reg. No. 39,435; my patent attorneys, and Peter Lam, Reg. No. P44,855; Thomas Raleigh Lane, Reg. No. 42,781; Gene I. Su, Reg. No. 45,140; and Calvin E. Wells, Reg. No. P43,256; my patent agents, of INTEL CORPORATION with full power of substitution and revocation, to prosecute this application and to transact all business in the Patent and Trademark Office

Send correspondence to Thomas M. Coester, Reg. No. 39,637, BLAKELY, SOKOLOFF, TAYLOR &

(Name of Attorney or Agent)

ZAFMAN LLP, 12400 Wilshire Boulevard, 7th Floor, Los Angeles, California 90025 and direct telephone calls to Thomas M. Coester, Reg. No. 39,637, (310) 207-3800.

(Name of Attorney or Agent)

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Full Name of Sole/First Inventor (given name, family name) Narsing K. Vijayrao

Inventor's Signature N.K. Vijayrao Date 22 June, 2000

Residence Santa Clara, CA Citizenship India
(City, State) (Country)

P. O. Address 1901 Halford Avenue, Apt. #103
Santa Clara, CA 95051 USA

Full Name of Second/Joint Inventor (given name, family name) Chi Keung Lee

Inventor's Signature [Signature] Date 22-June, 2000

Residence San Jose, CA Citizenship China
(City, State) (Country)

P. O. Address 1523 Briartree Drive

San Jose, CA 95131 USA

Full Name of Third/Joint Inventor (given name, family name) Sudarshan Kumar

Inventor's Signature Sudarshan Kumar Date 22 June, 2000

Residence Fremont, CA Citizenship India
(City, State) (Country)

P. O. Address 1300 Camero Way

Fremont, CA 94539 USA

Full Name of Fourth/Joint Inventor (given name, family name) _____

Inventor's Signature _____ Date _____

Residence _____ Citizenship _____
(City, State) (Country)

P. O. Address _____

Full Name of Fifth/Joint Inventor (given name, family name) _____

Inventor's Signature _____ Date _____

Residence _____ Citizenship _____
(City, State) (Country)

P. O. Address _____